APPLICATION UNDER UNITED STATES PATENT LAWS

Atty. Dkt. No. 202102-272220
(C/M#)

Invention: ENCRYPTION KEY DISTRIBUTION AND NETWORK REGISTRATION SYSTEM, APPARATUS AND METHOD

Inventor (s): Stephan Walter Gehring, Daniel Paul Peters, Jason Lee Ellis and Satish Ananthakrishnan

Pillsbury Winthrop LLP
Intellectual Property Group
50 Fremont Street
P.O. Box 7880
San Francisco, CA 94105-2228
Attorneys
Telephone: (415) 983-1000

This is a:

- ☐ Provisional Application
- ☒ Regular Utility Application
- ☐ Continuing Application
  - ☐ The contents of the parent are incorporated by reference
- ☐ PCT National Phase Application
- ☐ Design Application
- ☐ Reissue Application
- ☐ Plant Application
- ☐ Substitute Specification
  Sub. Spec Filed _____
  in App. No. ___/_____
- ☐ Marked up Specification re
  Sub. Spec. filed _____
  In App. No ___/_____

Certificate of Express Mailing Under 37 C.F.R. §1.10

I hereby certify that this correspondence (along with any paper referred to as being attached or enclosed) is being mailed via "Express Mail Post Office to Addressee" service of the United States Postal Services (Express Mail Label No. EL 920 299 226 US) on the date shown below in an envelope addressed to the Assistant Commissioner for Patents, U.S Patent and Trademark Office, Washington, D.C 20231

Dated: December 21, 2001

By: K A Cabello
Kim Cabello

SPECIFICATION

ENCRYPTION KEY DISTRIBUTION AND NETWORK REGISTRATION SYSTEM, APPARATUS AND METHOD

BACKGROUND

Related Applications

This application claims the benefit of co-pending U.S. Provisional Application Serial No. 60/314,145 filed August 22, 2001.

Field of the Invention

Aspects of the present invention relate in general to the field of communications network access security.

Background

The broadcast nature of wireless data transmission makes transmitted information susceptible to interception by third parties. Wireless networks, just like public networks, such as the Internet, must hence be considered insecure. As information transferred wirelessly between electronic devices is often of proprietary nature, there is a need to protect this information from eavesdropping.

Conventionally, one method of securing network access is by using encryption. In a typical system using encryption, a sender encrypts the information using an encryption engine and an encryption key. The resulting information is rendered unintelligible to any party except to the recipient in possession of the decryption key. The recipient decrypts the received information using a decryption engine and the appropriate decryption key and thus translates the encrypted information into its original readable form.

Traditionally, the same key is used for both encryption and decryption. This is called "private key" or "symmetric" cryptography. Keys that are used for both encryption and decryption are referred to as "symmetric keys." The shared key must remain secret since any party in possession of this key can decrypt information previously encrypted with the key. In order to establish a shared secret key, the key must be distributed among the communicating parties by means of a secure channel.

In public-key cryptography, the encryption key and the decryption key are distinct. The encryption key is public and can therefore be sent to the other party over an insecure communication channel. The decryption key is kept private and never revealed. To encrypt a message for a recipient, the sender uses the recipient's public encryption key. The encrypted message is then sent to the recipient over the insecure channel. The recipient uses its private decryption key to decrypt the received encrypted message.

BRIEF DESCRIPTION OF THE DRAWINGS

FIGS. 1A and 1B illustrate embodiments of a system to securely distribute a shared secret network encryption key from a host to a wireless peripheral device.

FIG. 2 is a block diagram of a wireless module to securely distribute a shared secret network encryption key from a host to a wireless peripheral device.

FIG. 3 is a block diagram of a media access control (MAC) layer to securely distribute a shared secret network encryption key from a host to a wireless peripheral device.

FIGS. 4A-4C are block diagrams of alternate embodiments of a key management and encryption unit (KMEU) to securely distribute a shared secret network encryption key from a host to a wireless peripheral device.

FIG. 5 is a flowchart of a host method 500, an embodiment to securely distribute a shared secret network encryption key from a host to a wireless peripheral device.

FIG. 6 depicts a flowchart of a peripheral method 600, a wireless peripheral device embodiment to securely receive a shared secret network encryption key from a host.

FIG. 7 illustrates a flowchart of a peripheral method 700, an alternate embodiment to securely receive a shared secret network encryption key from a host to a wireless peripheral device.

FIG. 8 is a flowchart of a peripheral method 800, an alternate embodiment to securely receive a shared secret network encryption key from a host to a wireless peripheral device.

DETAILED DESCRIPTION

The primary advantage of public-key cryptography is that the secret private key need never be revealed to any party. However, public-key methods can be significantly slower than established private-key methods, which is an important drawback for high-speed communication with devices with very limited computational capabilities such as computer peripherals. One aspect of the invention is taking advantage of the discovery that public-key methods are unsuited for broadcasting encrypted information to multiple recipients since each recipient device has its own decryption key, whereas private-key methods use a single shared decryption key.

Consequently, another aspect of the present invention is the discovery that there is therefore a need to securely distribute a shared secret encryption key among communicating parties.

Aspects of the present invention further include a system, apparatus and method to securely distribute a shared secret network encryption key from a host to a wireless peripheral device. In some aspects, the sharing of the secret network encryption key is accomplished without the intervention of a man-machine interface. Embodiments enable a wireless device to receive and store multiple encryption keys, to select the appropriate encryption key depending on the network in range of the device, and thus to freely associate with multiple networks.

It is understood that although the following embodiments are disclosed using a wireless connection to communicate between the host device and peripheral devices, other embodiments may equally apply to other forms of communicating information between devices. Other equally applicable media for communicating include, but are not limited to, wired connections, optically coupled connections, and any other connection as is known in the art.

FIG. 1A is a simplified diagram depicting system 100, constructed and operative in accordance with an embodiment of the present invention. System 100 is configured to securely distribute a shared secret network encryption key from a wireless host device 10 to a wireless peripheral device 20.

In some embodiments, wireless peripheral devices 20 are expected to be limited in resources, lacking a man-machine interface for network-related tasks. These network-related tasks include entering peripheral device encryption keys. Also, in many embodiments, wireless peripheral devices 20 may not have a processor available to perform complex communication tasks.

It is understood that method embodiments of the present invention may securely distribute a shared secret network encryption key from a host device 10 to any number of wireless peripheral devices 20a-*n*, although some embodiments using particular data

protocols may be limited in number of peripheral devices 20, by the data protocol itself. Furthermore, it is understood that the embodiments described herein may communicate with any wired or wireless communications protocol and technology known in the art. Examples of wireless communication protocols and technology include, but are not limited to: wireless Universal Serial Bus (wireless USB), ultra-wide-band (UWB), wireless Institute of Electrical and Electronics Engineers (IEEE) Standard No. 1394 ('wireless Firewire'), Institute of Electrical and Electronics Engineers (IEEE) Standard Nos. 802.11a and 802.11b ("Wi-Fi"), Bluetooth, and wireless RS-232. For convenience, embodiments below will now be discussed using the Universal Serial Bus (USB) protocol.

As shown in FIG. 1B, constructed and operative in accordance with an embodiment of the present invention, the wireless devices 10, 20 depicted in system 100 may comprise host 15 and peripheral devices 25 coupled to a wireless module 110 embodiment of the present invention. For example, wireless host device 10 may be further comprised of host 15 and a wireless module 110*x*.

Wireless modules 110a-x are any devices that allow a conventional computing device to communicate with other wireless modules 110 or wireless devices 10, 20 using a method embodiment of the present invention, as described in the claims below. The host device 15 interacts with the wireless network through wireless module 110. The wireless module 110 sends data received from the host 15 to wireless peripheral devices 20 and passes on data received from wireless peripheral devices 20 through the network to the host 15.

FIG. 2 is a block diagram of wireless module 110, constructed and operative in accordance with an embodiment of the present invention. It is well understood by those knowledgeable in the art that the elements of FIG. 2 may be implemented as structures in

hardware, firmware, or as software instructions and data encoded on a computer-readable storage medium. Wireless module 110 comprises device interface 102, media access control layer (MAC) 104, physical layer (PHY) 106, and radio 108. In some embodiments, wireless module 110 may also include an antenna (not shown), as is known in the art.

Device interface 102 is any interface, as is known in the art, which allows wireless module 110 to interface with host 15 or peripheral device 25. In some embodiments, device interface 102 may emulate a standard communications protocol so that host 15 or peripheral device 25 does not realize that it is communicating with another wireless device. For example, in wireless USB embodiments, device interface 102 may emulate a USB interface.

Media access control layer 104 is any structure that governs access to the wireless transmission media. As shown in FIG. 2, media access control layer 104 is coupled to device interface 102 and physical layer 106, enabling device interface 102 to exchange data with the physical layer 106.

Physical layer 106 is any structure that provides the procedures involved in transferring a single bit across the wireless frequency spectrum being used by the wireless module 110.

Radio 108 may be any radio frequency (RF) transceiver known in the art to allow wireless module 110 to transmit and receive communication signals.

FIG. 3 is a block diagram of a media access control layer 104, constructed and operative in accordance with an embodiment of the present invention. In this embodiment, media access control layer 104 comprises a protocol management unit 300 and a key management and encryption unit (KMEU) 400.

Protocol management unit 300 is the structure that enables media access control layer 104 to communicate using a particular wireless communications protocol, as described above. For example, in a wireless Universal Serial Bus embodiment, protocol management unit 300 (wireless USB) processes messages from device interface 102 and regulates the message access to the physical layer 106 using the Universal Serial Bus protocol. In embodiments using different wireless protocols, a corresponding protocol may be implemented in protocol management unit 300.

Encryption and decryption of messages may be performed by key management and encryption unit 400. As shown in FIG. 3, KMEU 400 may be situated in the MAC layer 104. In some embodiments, however, media access control layer 104 may only comprise protocol management unit 300. In such embodiments, the key management and encryption unit 400 may be located in, or functionally distributed within the wireless module 110, among other parts of the path between devices 15 or 25 and the antenna. Therefore, the location of the key management and encryption unit 400 should not be viewed as limiting the scope of the embodiments of the present invention, which are limited solely by the claims.

FIGS. 4A-4C are block diagrams of alternate embodiments of a key management and encryption unit 400, constructed and operative in accordance with an embodiment of the present invention. FIGS. 4A and 4C illustrate host embodiments of a key management and encryption unit, 400A and 400C. FIG. 4B depicts a peripheral embodiment of a key management and encryption unit 400B.

In FIG. 4A, the host key management and encryption unit 400A comprises a decryption unit (DEC) 402, an encryption unit (ENC) 404, at least two key storage units 408a-b, a key selector 406, and a host device control unit 410.

Encryption unit 404 encrypts messages to be transmitted through the wireless network 100 using a symmetric (i.e. "private") encryption key.

Received messages are decrypted by the decryption unit 402 using the same key as the encryption unit 404. Encryption unit 404 and decryption unit 402 may use any symmetric encryption-decryption method known in the art, and thus the embodiments vary according to the selected encryption algorithm.

The host device encryption key storage 408a stores the shared host secret key used for communication within the network. The shared host secret key may also be referred to as the "host device encryption key." In general, the shared host secret key may be globally unique. In alternate embodiments, it is sufficient that the shared host secret key be distinct from all encryption keys being used in other networks within operable range of network 100.

In some embodiments, host device encryption key storage 408a may be permanent memory such as a non-volatile memory, Read-Only Memory (ROM), Programmable Read-Only Memory (PROM), Erasable Programmable Read-Only Memory (EPROM), Electronically Erasable Programmable Read-Only Memory (EEPROM), or other persistent data storage medium as is known in the art.

In other embodiments, where the wireless host device 10 is never turned off, host device encryption key storage 408a may be stored in a volatile form of read-write memory.

Temporary key storage 408b may be any memory device as is known in the art, including but not limited to, Random Access Memory (RAM), Erasable Programmable Read-Only Memory (EPROM), Electronically Erasable Programmable Read-Only Memory (EEPROM), registers, and any form of modifiable (volatile or nonvolatile) memory, and temporary or persistent memory storage. For example, temporary key

storage 408b may be a register built of D-Flip-flops or Erasable Programmable Read-Only Memory (EPROM).

The host device control unit 410 may be any form of intelligence that instructs selector 406 on selecting a key storage unit 408. In some embodiments host device control unit 410 may be a finite state machine. In yet other embodiments, host device control unit 410 may be a microprocessor executing instructions fetched from an instruction memory or other computer-readable medium.

Under the control of the host device control unit 410, the key used for encryption and decryption may be selected to be either a host device encryption key storage 408a or a temporary key provided by the host in temporary key storage 408b. Key selection is performed by selector 406. In some embodiments, selector 406 may be implemented as a multiplexer, although embodiments may use any selection implementation known in the art.

FIG. 4C depicts an alternate embodiment of the host key management and encryption unit 400C. In this embodiment, an additional selector 406b chooses between the host encryption key storage unit 408a and messages received from wireless host device 10, constructed and operative in accordance with an embodiment of the present invention.

In FIG. 4B, the peripheral key management and encryption unit 400B comprises a decryption unit (DEC) 402, an encryption unit (ENC) 404, a plurality of encryption key storage units 409a-*n*, a key selector 406, and a peripheral device control unit 411.

Under the control of the peripheral device control unit 411, the key used for encryption and decryption may be selected from the set of encryption key storage units 409a-*n*. The set of encryption key storage units includes a peripheral device encryption key storage unit 409a and one or more shared secret key storage units 409b-*n*. Secret keys

may be received through the wireless network 100 from one or more wireless host devices 10. Key selection is performed by selector 406.

The implementation of encryption unit 404 and decryption unit 402 may be identical to that of the host key management and encryption unit 400A in that encryption unit 404 encrypts messages to be transmitted through the wireless network 100 using a symmetric (i.e. "private") encryption key. Decryption unit 402 uses the same key as the encryption unit 404 to decrypt the received messages. Encryption unit 404 and decryption unit 402 may use any symmetric encryption-decryption method known in the art, and thus the embodiments vary according to the selected encryption algorithm.

In some embodiments, selector 406 may be implemented as a multiplexer, although embodiments may use any selection implementation known in the art.

The peripheral device encryption key storage unit 409a stores the peripheral device encryption key used for shared host key distribution. In some embodiments, the peripheral device encryption key may be globally unique. A globally unique key precludes the possibility of two peripherals sharing the same secret key. In some alternate embodiments, it may be sufficient that the peripheral key be distinct from all encryption keys being used in other networks within operable range of network 100.

Peripheral control unit 411 may be any intelligence that aids selector 406 in the selection of one of the shared secret key storage units 409b-*n*.

The wireless peripheral device 20 interacts with the wireless network 100 through a peripheral embodiment of the wireless module 110. The wireless module 110 sends data received from the peripheral 25 to the wireless host device 10 and other wireless peripherals 20. Furthermore, wireless module 110 passes on data received from the wireless host 10 and other wireless peripherals 20a-*n* to the peripheral 25. In some embodiments, as described above, wireless module 110 may include an encryption

engine, such as a key management and encryption unit 400 and a unique secret encryption key. The encryption key is used to encrypt and decrypt a host encryption key sent from the wireless host device 10 to the wireless module 110 or wireless peripheral device 20 during configuration. The wireless module 110 or wireless peripheral device 20 can store one or more host device encryption keys in its key memory 409b-*n*.

The wireless peripheral device 20 can communicate with any wireless host 10 whose encryption key it has stored in memory. The host device encryption key is securely transferred from the wireless host device 10 to the wireless peripheral 20 or wireless module 110 using any of the method embodiments described herein.

FIG. 5 is a flowchart of a host method 500, an embodiment to securely distribute a shared secret network encryption key from a host 10 to a wireless peripheral device 20 or wireless module 110, constructed and operative in accordance with an embodiment of the present invention.

Initially, at block 502, the wireless host device 10 receives the peripheral device encryption key, which is input by the user. In some embodiments, this may be accomplished through device configuration application software, as is known in the art.

As part of block 502, wireless module 110 or wireless host device 10 stores the peripheral device encryption key. In some embodiments, the key management and encryption unit 400 stores the peripheral key received from the host in temporary key storage 408b.

The peripheral device encryption key may be delivered with the wireless peripheral device 20, printed on a separate piece of paper, attached as a label to the device, or other means known in the art to associate an encryption key with a wireless peripheral device 20. In some embodiments, wireless module 110 receives the peripheral device encryption key from the host 15 or receives the key as input by the user.

The wireless module 110 sets its encryption key to the received peripheral device encryption key, block 504. In some embodiments, host device control unit 410 and selector 406 guide the selection of the peripheral device encryption key stored in temporary key storage 408b as part of block 504.

A key distribution message containing the shared host secret key, which may be stored in host device encryption key storage 408a, is encrypted by the encryption unit 404 using the peripheral device encryption key, block 506.

This key distribution message is then transmitted to the wireless peripheral device 20 which stores the received host secret key, block 508.

The encryption key of the host device is then reset to the shared host secret key, which may be stored in host device encryption key storage 408a, at block 510, and communication with the wireless peripheral device 20 using the shared host secret key can commence.

An embodiment of a host method, implemented in a host device control unit 410, is shown below in Table 1.

```
valid = false;
currentKey = key[0];
while (true) {
    receiveFromHost(cmd, data, valid);
    if (valid) {
       if (cmd == SENDKEY) {              // send host key to peripheral
          key[1] = data;                  // store peripheral key
          currentKey = key[1];            // use peripheral's key
          sendToNetwork(SETKEY, key[0]);  // send host key (= key[0])
          currentKey = key[0];            // use host key again
       } else if (cmd == DATA) {          // send data to peripheral
          sendToNetwork(DATA, data);
       }
    }
}
```

TABLE 1. Host Device Control Unit Pseudo-Code

FIG. 6 depicts a flowchart of a peripheral method 600, an embodiment to securely receive a shared secret network encryption key from a host 10 to a wireless peripheral device 20 or wireless module 110, constructed and operative in accordance with an embodiment of the present invention.

When the wireless peripheral device 20 is turned on, or otherwise activated, the wireless peripheral device 20 listens for key distribution messages encoded with its own peripheral device encryption key, block 602. As discussed above, in some embodiments the peripheral device encryption key may be stored in a peripheral device encryption key storage unit 409a. The wireless peripheral device 20 selects the peripheral device encryption key as the encryption (and decryption) key and listens for key distribution messages.

When the wireless peripheral device 20 receives a key distribution message from a host 10, at block 604, decryption unit 402 decrypts the message and retrieves the shared host secret key, block 606.

An unused peripheral device encryption key storage unit 409b-*n* is selected, block 608, and the decoded shared host secret key (also referred to as the host device encryption key) is stored within the peripheral device encryption key storage unit 409*n*, block 610.

Thereafter, whenever the peripheral wishes to communicate with a specific host device 10, it sets the encryption key to the corresponding shared host secret key stored within the peripheral device encryption key storage 409*n*, block 612, and communicates with the host device 10, block 614.

FIG. 7 illustrates a flowchart of a peripheral method 700, an alternate embodiment to securely receive a shared secret network encryption key from a wireless host device 10, constructed and operative in accordance with an embodiment of the present invention.

When the wireless peripheral device 20 is activated, the wireless peripheral device 20 listens for key distribution messages encoded with its own peripheral device encryption key, block 602. In some embodiments the peripheral device encryption key may be stored in a peripheral device encryption key storage 409a. The wireless peripheral device 20 selects the peripheral device encryption key as the encryption (and decryption) key and listens for key distribution messages. Process 700 proceeds to decision block 702.

At decision block 702, peripheral control unit 411 determines whether a key distribution message has been sent from the wireless host device 10.

When the wireless peripheral device 20 receives a key distribution message, as determined by decision block 702, decryption unit 402 decrypts the message and retrieves the shared host secret key , block 606. An unused peripheral device encryption key storage unit 409b-*n* is selected, block 608, and the decoded shared host secret key (also known as the host device encryption key) is stored within the peripheral device encryption key storage 409*n*, block 610. Thereafter, whenever the peripheral wishes to communicate to a specific host device 10, it sets the encryption key to the corresponding shared host secret key stored within the peripheral device encryption key storage 409*n*, block 612, and communicates with the host device 10, block 614. Process 700 then returns to block 602.

If a distribution message has been sent from the wireless host device 10, as determined by decision block 702, process 700 continues to decision block 704.

At decision block 704, process 700 determines whether other network traffic is detected. If no traffic is detected, process 700 returns to block 602. If other network traffic is detected, process 700 continues at block 706, and peripheral control unit 411 attempts to decode the message using another encryption key stored in one of the

peripheral device encryption key storage units 409b-n. If the network traffic matches a stored key, as determined by decision block 708, communication with a host device is continued with the stored host device encryption key. Otherwise, if the network traffic does not match, process 700 tries the next key, block 706, until all stored keys are attempted, block 710, and then process 700 returns to block 602.

FIG. 8 is a flowchart of a peripheral process 800, an alternate embodiment to securely receive a shared secret network encryption key from a wireless host device 10 to a wireless peripheral device 20 or wireless module 110, constructed and operative in accordance with an embodiment of the present invention. Process 800 allows a wireless peripheral device 20 or wireless module 110 to change the selected encryption key, when associating with one wireless network 100 to another.

Initially at block 802, the wireless peripheral device 20 selects the peripheral device encryption key stored in peripheral device encryption key storage 409a.

Any key timeout situation is cleared at block 804.

Wireless module 110 or wireless peripheral device 20 listens and receives a message over the wireless network at block 806.

At block 808, process 800 determines whether the received message is valid. A valid message is one that has been received without transmission error, and that can be successfully decrypted by wireless module 110. In some embodiments of the invention, validity is determined on whether decryption unit 402 is able to decrypt the received message with the selected encryption key. If the received message is valid, as determined by decision block 808, flow continues at block 816. Otherwise, if the received message is invalid, process flow continues at block 810.

At block 810, process 800 determines whether there has been a message timeout. If no message has been successfully received after a predetermined amount of time, a

timeout occurs. If there is no timeout, flow returns to block 806. If there is a message timeout, as determined by block 810, decision block 812 checks to see if all stored encryption keys have been examined. If all the encryption keys are examined, the process begins anew and flow returns to block 802. If all the encryption keys have not been examined, the next available key is selected at block 814, and flow continues at block 804.

At block 816, process 800 checks to see if the valid message is a key distribution message. If the message is a set encryption key message, as determined by decision block 816, flow continues at block 818. Otherwise, flow continues at decision block 824.

At block 818, the peripheral control unit 411 selects a peripheral key storage unit 409 and stores the received encryption key in the selected storage unit 409. In selecting a storage unit 409, the peripheral control unit 411 first determines whether there are peripheral key storage units 409b-*n* that are not in use. An unused storage unit 409 is then selected.

In some embodiments, if all of the peripheral key storage units 409b-*n* are already in use, the peripheral chooses the last peripheral key storage units 409*n*. Other key replacement strategies as are known in the art, can be equally applied.

The received encryption key is then stored in the peripheral key storage unit 409, block 820. The next available key is selected, and process flow returns to block 804.

If the message is not a set encryption key message, as determined by decision block 816, decision block 824 determines whether the message is a data message. If the message received is a data message, the message is forwarded to the peripheral device 25 connected to the wireless module 110 at block 826. Otherwise, flow returns to block 804.

An embodiment of a peripheral method, implemented in a peripheral device control unit 411, is shown below in Table 2.

```
MAXKEYS = N;              // N >= 2
nofKeys = 1;
keyIndex = 0;
currentKey = key[keyIndex];
valid = timeout = false;
while (true) {
    receiveFromNetwork(cmd, data, valid, timeout);
    if (valid) {          // received a packet
       if (cmd == SETKEY) {                // add new network key
          if (nofKeys < MAXKEYS) {         // not full, append
             keyIndex = nofKeys;
             nofKeys = nofKeys + 1;
          } else          // full, replace last key
             keyIndex = nofKeys - 1;
          key[keyIndex] = data;            // store new key (= data)
          currentKey = key[keyIndex];
       } else if (cmd == DATA) {           // received data packet
          sendToPeripheral(DATA, data);    // pass data on
       }
    } else if (timeout) {                  // no response in a while
       if (keyIndex < nofKeys - 1)         // try next used key
          keyIndex = keyIndex + 1;
       else               // wrap around to first key
          keyIndex = 0;
       currentKey = key[keyIndex];
    }
}
```

TABLE 2. Pseudo-Code for Peripheral Device Control Unit

The previous description of the embodiments is provided to enable any person skilled in the art to practice the invention. The various modifications to these embodiments will be readily apparent to those skilled in the art, and the generic principles defined herein may be applied to other embodiments without the use of inventive faculty. Thus, the present invention is not intended to be limited to the embodiments shown herein, but is to be accorded the widest scope consistent with the principles and novel features disclosed herein.

WHAT IS CLAIMED IS: